

Fachcurriculum Informatik 4. Klasse

Der Informatikunterricht trägt dazu bei, digitale Hilfsmittel sachgerecht und zielgerichtet, verantwortungs- und sinnvoll einzusetzen. Sie sind im Informatikunterricht stets Medium, Werkzeug und Inhalt des Lernens zugleich. Schülerinnen und Schüler lernen die wichtigsten theoretischen Grundlagen der Informationswissenschaften und erwerben Fertigkeiten in der Nutzung von Instrumenten der Informatik. Dabei sollen ihnen die Prinzipien klar werden, auf denen die verschiedenen aktuellen Varianten eines Informatiksystems gründen. Schülerinnen und Schüler lernen neben der Bedienung entsprechender Systeme vor allem die darauf ausgerichteten Arbeitsmethoden erkennen und beherrschen, damit sie auch künftige technische Entwicklungen besser beurteilen können und ihre Vorkenntnisse und bestehenden Fähigkeiten selbstständig weiterentwickeln. Im Informatikunterricht sind die inhaltlichen Bereiche Information und Daten, Algorithmen, Sprachen und Automaten, Informatiksysteme, Informatik, Mensch und Gesellschaft immer eng verzahnt mit den Prozessbereichen des Modellierens und Implementierens, Begründens und Bewertens, Strukturierens und Vernetzens, Kommunizierens und Kooperierens, Darstellens und Interpretierens. Die Schülerinnen und Schüler erwerben die notwendigen Kompetenzen, um die durch Informatiksysteme veränderte Lebenswelt zumindest zu einem wesentlichen Teil zu durchschauen, verstehen, beurteilen und mitgestalten zu können. Im Informatikunterricht werden die Gemeinsamkeiten der für alle Disziplinen gültigen informatischen Strukturen und Methoden einschließlich der fachlichen Begriffswelt herausgearbeitet. Deshalb wird der Informatikunterricht fächerübergreifend und fächerverbindend, hauptsächlich zur Mathematik, Physik und zu den Naturwissenschaften gestaltet. Weiters ist es wichtig, Synergien mit dem Umfeld zu finden, indem die Zusammenarbeit mit Universitäten, Forschungseinrichtungen, Wissenschaftsmuseen und der Arbeitswelt ermöglicht wird.

Kompetenzen am Ende der 5. Klasse

Die Schülerin, der Schüler kann

- mit den gängigsten Software-Programmen umgehen und sie für das eigene Lernen und die Kommunikation im Netz zielgerichtet einsetzen
- unter Nutzung innovativer Methoden und Techniken Daten suchen, analysieren, interpretieren, organisieren, verarbeiten, darstellen und präsentieren
- zu gegebenen Sachverhalten informatische Modelle erstellen, diese mit geeigneten Werkzeugen implementieren und die Implementierung der Modelle reflektieren
- Möglichkeiten und Grenzen der Informatik innerhalb des kulturellen und sozialen Kontextes, in dem sie angewandt wird, bewusst wahrnehmen und reflektieren
- sich selbstständig neue Anwendungen und Informatiksysteme erschließen

Programmierung			
Fertigkeiten und Fähigkeiten	Kenntnisse	Inhalte	Methodisch-didaktische Hinweise
Die Konstrukte einer Programmiersprache beherrschen und Problemstellungen mittels ein Programmiersprache lösen können.	<p>Syntax und Semantik einer Programmiersprache</p> <p>Lösen von Problemen mittels einer Programmiersprache</p> <p>Entwickeln von Algorithmen in einem Pseudocode oder in einer Programmiersprache</p>	<p>Programmiersprachen</p> <ul style="list-style-type: none"> • Programmierparadigmen • Quell- und Maschinencode • Compiler und Interpreter • Debugger <p>Einführung in die Programmierung</p> <ul style="list-style-type: none"> • Von der Problemstellung zum Algorithmus (Ablaufdiagramm, Struktogramm, Pseudocode) • Ein- Ausgabe • Variablen und Konstanten • Datentypen • Operatoren • Error Handling • Verzweigungen • Schleifen • Funktionen • Casten • Arrays und Strings • Zufallszahlen • IDE (Dev-C++, CLion, Atom) 	<p>Lehrervortrag</p> <p>Programmierübungen Programmierung grundlegender Algorithmen</p> <p>Algorithmen, welche aus der Mathematik bekannt sind, werden in eine Programmiersprache übertragen.</p>

Minimalziele Themenbereich: Programmierung

- Die Funktionsweise der wichtigsten Konstrukte einer Programmiersprache, wie z.B. Variable, Array, Schleife, Verzweigung, Funktion, erklären können.
- Den Output von Algorithmen vorhersagen können.
- Einfache semantische und syntaktische Fehler erkennen und beheben können.
- Einfache Problemstellungen mittels einer Programmiersprache lösen können.

Algorithmen			
Fertigkeiten und Fähigkeiten	Kenntnisse	Inhalte	Methodisch-didaktische Hinweise
einen Algorithmus in Pseudocode und/oder in einer bestimmten Programmiersprache entwickeln	algorithmische Grundbausteine Problemlösung mittels eines Algorithmus und einer Programmiersprache	Suchalgorithmen Einfache Sortieralgorithmen Komplexere Sortieralgorithmen	Lehrervortrag Programmierübungen Programmierung grundlegender Algorithmen Algorithmen, welche aus der Mathematik bekannt sind, werden in eine Programmiersprache übertragen.

Minimalziele Themenbereich: Algorithmen

- Einfache Algorithmen in Pseudocode und in einer bestimmten Programmiersprache entwickeln können.
- Den Output von Algorithmen vorhersagen können.
- Einfache semantische und syntaktische Fehler erkennen und beheben können.
- Eigene Algorithmen zur Problemlösung entwickeln können.

Netzwerke			
Fertigkeiten und Fähigkeiten	Kenntnisse	Inhalte	Methodisch-didaktische Hinweise
Der Schüler kennt den allgemeinen Aufbau eines Computernetzwerkes und kann ein Netzwerk konfigurieren.	<p>Der Schüler kennt die wichtigsten Geräte, welche in einem Heimnetzwerk verwendet werden</p> <p>Der Schüler kennt das ISO/OSI-Schichtenmodelle und dessen Zweck</p>	<p>Netzwerk Hardware Komponenten</p> <p>Netzwerkprotokolle</p> <p>ISO/OSI Referenzmodell</p> <p>TCP/IP Referenzmodell</p> <p>Adressierungsarten in einem Netzwerk</p>	<p>Lehrervortrag</p> <p>Unterlagen von Seiten der Lehrperson</p> <p>Übungen mit Simulationssoftware oder HW Komponenten</p>

Minimalziele Themenbereich: Netzwerke

- Der Schüler kennt die wichtigsten Geräte, welche in einem Heimnetzwerk verwendet werden und kann deren Aufgabe erklären
- Der Schüler kennt den allgemeinen Aufbau eines Computernetzwerkes und kann einfache Konfigurierungsproblemen lösen
- Der Schüler kennt das ISO/OSI-Schichtenmodelle und dessen Zweck